jurBERT: A Romanian BERT Model for Legal Judgement Prediction

Mihai Masala, Radu Iacob
University Politehnica of Bucharest,
313 Splaiul Independentei, 060042,
Bucharest, Romania
{mihai_dan.masala, radu.iacob}@upb.ro

Ana Sabina Uban, Marina Cidota
University of Bucharest,
14 Academiei, 010014
Bucharest, Romania
{ana.uban, marina.cidota}@gmail.com

Horia Velicu
BRD Groupe Societe Generale
1-7 Ion Mihalache, 0111171
Bucharest, Romania
horia.velicu@brd.ro

Traian Rebedea
University Politehnica of Bucharest,
313 Splaiul Independentei, 060042,
Bucharest, Romania
train.rebedea@upb.ro

Marius Popescu
University of Bucharest,
14 Academiei, 010014
Bucharest, Romania
marius.popescu@fmi.unibuc.ro

Abstract

Transformer-based models have become the de facto standard in the field of Natural Language Processing (NLP). By leveraging large unlabeled text corpora, they enable efficient transfer learning leading to state-of-the-art results on numerous NLP tasks. Nevertheless, for low resource languages and highly specialized tasks, transformer models tend to lag behind more classical approaches (e.g. SVM, LSTM) due to the lack of aforementioned corpora. In this paper we focus on the legal domain and we introduce a Romanian BERT model pre-trained on a large specialized corpus. Our model outperforms several strong baselines for legal judgement prediction on two different corpora consisting of cases from trials involving banks in Romania.

1 Introduction

In recent years, a paradigm shift stormed the entire NLP field. Transformer (Vaswani et al., 2017) blocks allowed the development of large models that efficiently exploit the power of transfer learning. Pre-training transformers on large unlabeled text data, followed by a fast fine-tuning step has become the de facto approach across the field. Moreover, transformer based architectures (Devlin et al., 2019; Liu et al., 2020; Yang et al., 2019; Radford et al., 2018, 2019; Brown et al., 2020; Zhang et al., 2019) have achieved state-of-the-art results on several generic NLP tasks ranging from natural language understanding (Wang et al., 2018, 2019; Lai et al., 2017), question answering (Rajpurkar et al., 2018; Reddy et al., 2019) to Text-to-SQL (Yu et al., 2018, 2019b,a). Nevertheless, for low resource languages or highly specialized domains, pre-trained language models tend to underperform in part due to the lack of pre-training data or due to the generic nature of these large corpora. For this reason, specific BERT models have been trained and developed for numerous lan-

Natural Legal Language Processing Workshop 2021, pages 86–94

guages such as French (Martin et al., 2020; Le et al., 2020), Dutch (de Vries et al., 2019; Delobelle et al., 2020), Romanian (Masala et al., 2020; Dumitrescu et al., 2020), Finish (Virtanen et al., 2019), Spanish (Cañete et al., 2020) and for highly specialized domains such as Science (Beltagy et al., 2019), Legal (Chalkidis et al., 2020) or Biomedical (Lee et al., 2019).

In this work we set out to investigate the possibility of adapting and applying BERT models for legal judgement prediction on a small, noisy dataset, in a low resource language (Romanian). The corpus we use is a realistic representation of the kind of machine-readable data that is available to practitioners in this specialized field. The data, provided by a Romanian bank, is composed of original lawsuit documents, and features the most frequent types of cases pertinent to the banking domain.

Our contributions can be summarized as follows:

- We publicly release the first, to the best of our knowledge, pre-trained BERT models[1] specialized for the Romanian juridical domain.
- We propose and extensively analyze a general methodology for applying BERT models on real world juridical cases.
- We obtain state-of-the-art results on a small, noisy, highly specialized industry-provided corpus.

2 Related Work

The legal domain provides a wide range of different tasks in which NLP techniques can and have been used. Such tasks include detection of argumentative sentences (Moens et al., 2007; Palau and Moens, 2009), report summarization (Hachey and Grover, 2006; Galgani et al., 2012) or identification of the law areas that are relevant to a case (Boella et al., 2011; Şulea et al., 2017; Sulea et al., 2017).

Sulea et al. (2017) propose the usage of an ensemble of Support Vector Machines (SVMs) on word unigram and bigrams to solve three tasks related to French Supreme Court cases: predicting the law area of a case, predicting case ruling and estimating the time span of a given case or ruling. Similarly, Medvedeva et al. (2018) use SVMs on textual features (mainly word n-grams) to analyze and predict cases from the European Court of Human Rights. Katz et al. (2017) use random forest classifiers over handcrafted features (based rather on the context of the case than on the textual arguments) to predict the ruling of the Supreme Court of the United States. Chalkidis et al. (2020) investigate the usage of BERT models on multiple legal corpora. Their experiments show that further fine-tuning a general BERT model or training one from scratch on juridical data produces state-of-the-art results for legal text classification tasks. While both strategies are valid and the best one might depend on the given task, in our work we decide to pre-train a BERT model from scratch as we employ a significantly larger pre-training corpus (with a raw size of 160GB compared to only 12GB collected by Chalkidis et al. (2020)).

For small and noisy data, such as the real world BRDCases dataset we use in our work, large models may underperform compared to simpler models (Ezen-Can, 2020; Lai et al., 2021). Lately, string kernels (Lodhi et al., 2000, 2002), an efficient character-level comparison technique, have been used with promising results in low resource settings such as native language identification (Ionescu et al., 2016), dialect identification (Butnaru and Ionescu, 2018, 2019), chat understanding (Masala et al., 2018) or automated essay scoring (Cozma et al., 2018).

Dataset	Scope	Entries	Size
RoJur	pre-training	11M	-
RoBanking	downstream	108K	2,212 / 1,309
BRDCases	downstream	149	12,119 / 12,090

Table 1: Dataset statistics. Size is presented in terms of jurBERT tokens with mean and median values separated by /.

3 Datasets

The first dataset we employ, RoJur, comprises all the final rulings, containing both civil and criminal cases, published by any Romanian civil court between 2010 and 2018. Each sample contains: a description of the involved parties, a summary of the critical arguments made by the plaintiffs and the defendants, the legal reasoning behind the verdict and the final verdict itself. The names of the entities involved, as well as other identification details are anonymized throughout the document. Notably, the document is written by a human expert (i.e. the judge presiding over the case) who may have

[1] https://huggingface.co/readerbench

Model	MLM Acc	NSP Acc
jurBERT-base	89.36	99.23
jurBERT-large	90.05	99.29

Table 2: jurBERT NSP and MLM performance on the evaluation corpus

restructured or rephrased the original arguments made by the involved parties. We note that RoJur is a private corpus that can be rented for a significant fee.

We devise a second dataset, RoBanking, from rulings encountered in RoJur. Specifically, we extract common types of cases pertinent to the banking domain (e.g. administration fee litigations, enforcement appeals). From each ruling we only keep the summary of the arguments provided by the plaintiffs and the defendants, and a boolean value denoting which party was favoured in the final verdict.

Finally, we use BRDCases, representing a collection of cases in which a particular Romanian bank (BRD Groupe Société Générale Romania) was directly involved. Each sample contains a section with the arguments provided by the plaintiff and a section for those provided by the defendant. The content of each section is extracted from the original lawsuit files. The plaintiff section is obtained through an OCR process and by employing heuristics to remove content that may be irrelevant to the case. Consequently, the text is likely to contain typographical errors and other artifacts. Moreover, there may be significant differences in writing style, stemming from the possible gap in juridical knowledge between the involved parties. However, this type of input is a realistic representation of the machine readable data that is available to the attorneys handling a specific case in a Romanian bank.

Statistics pertaining to each dataset are presented in Table 1. The size of RoJur (160 GB as stored on disk) enabled us to pretrain a BERT model from scratch for the Romanian juridical domain. The remaining datasets, RoBanking and BRDCases, were used for downstream applications.

4 Model - jurBERT

For all intents and purposes we stick to the same model architecture and training procedure proposed by Devlin et al. (2019). We opt to train two variants of jurBERT, namely jurBERT-base and jurBERT-large, with Whole Word Masking (WWM), each with the same vocabulary of 33k tokens, for 40 epochs on a v3-8 TPU (kindly provided by Tensorflow Research Cloud[2]). For efficiency reasons we train with sequence lengths of 128 for 90% of the training steps, while for the last 10% of steps we use sequence lengths of 512. Evaluation results on the pre-training corpus, RoJur, are depicted in Table 2.

5 Evaluation

We evaluate our pre-trained model on the task of predicting whether the final verdict in a legal case is favourable to the plaintiff or the defendant. To this end, we leverage RoBanking and BRDCases. Despite the similarity in structure, there are significant differences between the two datasets, as presented in Section 3.

We extensively explore different fine-tuning strategies, to enable efficient transfer learning and to prevent catastrophic forgetting. Inspired by previous approaches (Araci, 2019; Howard and Ruder, 2018; Sun et al., 2019) we investigate several strategies for dealing with long texts (e.g. using the first, middle or last part of the text), pooling type (i.e. <CLS>, mean or max), layer unfreezing (e.g. optimize all weights all throughout the training process, gradual unfreezing of layers), learning rate (i.e. constant, discriminative or slanted triangular learning rate), dropout value, final fully connected layers (sizes and numbers) and different combinations of mentioned strategies. We note that the setup for finding the best training strategy is iterative: we test all aspects of a given strategy, select the best and only then moving to the next step while retaining previous strategies. Henceforth, we refer to the best strategy[3] as the *optimized* strategy.

Both downstream tasks are framed as binary classification tasks (i.e. given the arguments, who wins the case). The results are reported using k-fold cross validation, with 5 folds for RoBanking and 10 folds for BRDCases. Cross-entropy loss is minimized using the Adam optimizer (Kingma and Ba, 2015) as each model is trained 3 times. Finally we report the mean AUC and the standard deviation for each model.

6 Results

The results on RoBanking, using only the plaintiff's plea, are presented in Table 3. The upper

[2]https://www.tensorflow.org/tfrc

[3]More details about the process of finding the best and final training configuration can be found in Appendix A

Model	Strategy	Mean AUC	Std AUC
CNN	-	79.60	*
BI-LSTM	-	80.99	0.26
RoBERT-small	classic	68.81	0.13
RoBERT-base	classic	78.52	0.09
RoBERT-large	classic	79.43	0.28
jurBERT-base	classic	81.01	0.19
jurBERT-large	classic	80.38	0.32
RoBERT-small	optimized	70.54	0.28
RoBERT-base	optimized	79.74	0.21
+ handcrafted	-	79.82	0.11
RoBERT-large	optimized	76.53	5.43
jurBERT-base	optimized	**81.47**	0.18
+ handcrafted	-	81.40	0.18
jurBERT-large	optimized	78.38	1.77

Table 3: Results on RoBanking using only the plea of the plaintiff.

Model	Strategy	Mean AUC	Std AUC
BI-LSTM	-	84.60	0.59
RoBERT-base	optimized	84.40	0.26
+ handcrafted	-	84.43	0.15
jurBERT-base	optimized	86.63	0.23
+ handcrafted	-	**86.73**	0.22
jurBERT-large	classic	82.04	0.64

Table 4: Results on RoBanking using pleas from both the plaintiff and defendant.

half of Table 3 introduces the considered baselines, namely two standard CNN and BI-LSTM models with an attention mechanism, followed by three variants of a state-of-the-art Romanian BERT model, RoBERT (Masala et al., 2020). More details regarding the baselines can be found in Appendix B. Lastly, we introduce our proposed model with its two variants. Note that for the upper half of Table 3 we use a classic finetuning strategy as proposed by Devlin et al. (2019). In the lower half of Table 3 we present results using the best finetuning strategy. First, we notice jurBERT consistently outperforms the considered baselines in any setting. One interesting observation is that while jurBERT-large outperforms its *base* counterpart on the NSP and MLM tasks, it lags behind on downstream task performance irrespective of training strategy. Finally, incorporating the defendant's plea, leads to significant improvements for all considered models, as can be seen in Table 4.

In Table 5 we present the results on BRDCases. As this dataset is rather small and contains a significant amount of noisy data and very long texts, the challenge posed is significantly harder than in the case of RoBanking. Therefore, we notice the lower overall AUC score compared to the results for RoBanking. As this dataset is extremely small

Model	Mean AUC	Std AUC
SVM with SK	57.72	2.15
jurBERT-base†*	53.65	*
RoBERT-base	52.24	0.33
jurBERT-base	55.35	0.37
RoBERT-base†	53.24	1.76
+ handcrafted†	55.40	0.96
jurBERT-base†	59.65	1.16
+ handcrafted†	**61.46**	1.76

Table 5: Results on BRDCases. † denotes models that were first finetuned on RoBanking.* marks models with no further training on BRDCases, inference-only.

(only 149 entries) we introduce a simple Support Vector Machine (SVM) with string kernels as baseline. More details about the configuration used for the baseline can be found in Appendix B. In the first part of Table 5 we also present the model fine-tuned on RoBanking without further training on BRDCases. In the second half of Table 5 we present the results obtained by fine-tuning two different models on BRDCases, one only pre-trained and one pre-trained and further fine-tuned on RoBanking. While the two corpora differ in essence, fine-tuning on RoBanking greatly improves the downstream performance on BRDCases. Finally, we note the importance of the pre-training step (on RoJur corpus) as jurBERT consistently and significantly outperforms RoBERT for all considered models and experiments, but especially for the real-word use case (Table 5).

Lastly, we investigate the effectiveness of simple handcrafted features for the legal judgement prediction task. Handcrafted features include the county and the year for each case and, while already present in text, they were also added in the final decision layer in categorical form (one-hot encoding). Experiments with said features are marked accordingly (+ *handcrafted*) in Tables 3,4 and 5. In the case of RoBanking, the added features are not especially relevant, yielding mixed results: same or worse mean AUC when using only the plaintiff's plea; see Table 3), and slightly better results overall when using both the plaintiff's and the defendant's pleas (see Table 4). However, for BRDCases, handcrafted features provide a consistent improvement of around 2% absolute value for both RoBERT and jurBERT models (see Table 5). Our best model, jurBERT with added handcrafted features significantly outperforms the considered baseline with an almost 4% absolute value mean AUC increase.

7 Conclusions

In this work, to the best of our knowledge, we employed the first study on applicability of state-of-the-art NLP methods for Romanian legal judgement prediction. We pre-trained, released and evaluated our models with promising results on two highly practical datasets, RoBanking and BRD-Cases. On the first dataset, that contains a human-generated summary of key arguments, our model, jurBERT, outperforms the considered baselines. Turning to the second dataset, that contains all the original arguments of the involved parties, jurBERT is just slighty better than much simpler models, as it struggles to handle such long texts. Especially in this case, the limitations of BERT-like models with regards to the maximum input size are a significant factor that hampers their performance.

Our proposed methodology for legal judgment prediction on real world cases involves three steps. The first step is pre-training a BERT model on a general purpose collection of cases (in our case RoJur). The second step includes further training on a subset of the previous corpus (in our case RoBanking), in which the model learns to predict the verdict having access only to the summarized arguments of the involved parties. The final and the most important step in our work is training and evaluating on the industry-provided cases. One of our key findings is that the second step in this methodology is crucial for obtaining good results for legal judgment prediction. We emphasize that this methodology is language independent and can easily be applied to similar tasks.

Major improvement areas for our approach are the development and integration of more refined handcrafted features (e.g. the type of court or the identity of the judge) and tackling the problem of long texts that greatly exceed the maximum input size of our model. For the latter, lines of research include summarization of long texts or employing methods of increasing the inherent sequence length limit of transformer models (Zaheer et al., 2020; Beltagy et al., 2020).

References

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Guido Boella, Luigi Di Caro, and Llio Humphreys. 2011. Using classification to support legal knowledge engineers in the eunomos legal document management system. In *Fifth international workshop on Juris-informatics (JURISIN)*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Andrei Butnaru and Radu Tudor Ionescu. 2019. MOROCO: The Moldavian and Romanian dialectal corpus. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 688–698, Florence, Italy. Association for Computational Linguistics.

Andrei M. Butnaru and Radu Tudor Ionescu. 2018. Unibuckernel reloaded: First place in arabic dialect identification for the second year in a row. *CoRR*, abs/1805.04876.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Mădălina Cozma, Andrei Butnaru, and Radu Tudor Ionescu. 2018. Automated essay scoring with string kernels and word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 503–509, Melbourne, Australia. Association for Computational Linguistics.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. *arXiv preprint arXiv:2001.06286*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Stefan Dumitrescu, Andrei-Marius Avram, and Sampo Pyysalo. 2020. The birth of Romanian BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4324–4328, Online. Association for Computational Linguistics.

Aysu Ezen-Can. 2020. A comparison of lstm and bert for small corpus. *arXiv preprint arXiv:2009.05451*.

Filippo Galgani, Paul Compton, and Achim Hoffmann. 2012. Combining different summarization techniques for legal text. In *Proceedings of the workshop on innovative hybrid approaches to the processing of textual data*, pages 115–123.

Ben Hachey and Claire Grover. 2006. Extractive summarisation of legal texts. *Artificial Intelligence and Law*, 14(4):305–345.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Radu Tudor Ionescu, Marius Popescu, and Aoife Cahill. 2016. String kernels for native language identification: Insights from behind the curtains. *Computational Linguistics*, 42(3):491–525.

Daniel Martin Katz, Michael J. Bommarito, II, and Josh Blackman. 2017. A general approach for predicting the behavior of the supreme court of the united states. *PLOS ONE*, 12(4):1–18.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Tuan Lai, Heng Ji, and ChengXiang Zhai. 2021. BERT might be overkill: A tiny but effective biomedical entity linker based on residual convolutional neural networks. *CoRR*, abs/2109.02237.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. RoBERTa: A robustly optimized BERT pretraining approach.

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *J. Mach. Learn. Res.*, 2:419–444.

Huma Lodhi, John Shawe-Taylor, Nello Cristianini, and Christopher J. C. H. Watkins. 2000. Text classification using string kernels. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pages 563–569. MIT Press.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Mihai Masala, Stefan Ruseti, and Mihai Dascalu. 2020. RoBERT – a Romanian BERT model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6626–6637, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Mihai Masala, Stefan Ruseti, Gabriel Gutu-Robu, Traian Rebedea, Mihai Dascalu, and Stefan Trausan-Matu. 2018. Help me understand this conversation: Methods of identifying implicit links between cscl contributions. In *Lifelong Technology-Enhanced Learning*, pages 482–496, Cham. Springer International Publishing.

Masha Medvedeva, Michel Vols, and Martijn Wieling. 2018. Judicial decisions of the european court of human rights: Looking into the crystal ball. In *Proceedings of the Conference on Empirical Legal Studies*.

Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. 2007. Automatic detection of arguments in legal texts. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ICAIL '07, page 225–230, New York, NY, USA. Association for Computing Machinery.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: The detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL '09, page 98–107, New York, NY, USA. Association for Computing Machinery.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Octavia-Maria Sulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu P Dinu, and Josef Van Genabith. 2017. Exploring the use of text classification in the legal domain. *arXiv preprint arXiv:1710.09306*.

Octavia-Maria Şulea, Marcos Zampieri, Mihaela Vela, and Josef van Genabith. 2017. Predicting the law area and decisions of French Supreme Court cases. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 716–722, Varna, Bulgaria. INCOMA Ltd.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics*, pages 194–206, Cham. Springer International Publishing.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: Bert for finnish. *arXiv preprint arXiv:1912.07076*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Tao Yu, Rui Zhang, Heyang Er, Suyi Li, Eric Xue, Bo Pang, Xi Victoria Lin, Yi Chern Tan, Tianze Shi, Zihan Li, Youxuan Jiang, Michihiro Yasunaga, Sungrok Shim, Tao Chen, Alexander Fabbri, Zifan Li, Luyao Chen, Yuwen Zhang, Shreya Dixit, Vincent Zhang, Caiming Xiong, Richard Socher, Walter Lasecki, and Dragomir Radev. 2019a. CoSQL: A conversational text-to-SQL challenge towards cross-domain natural language interfaces to databases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1962–1979, Hong Kong, China. Association for Computational Linguistics.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-SQL task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921, Brussels, Belgium. Association for Computational Linguistics.

Tao Yu, Rui Zhang, Michihiro Yasunaga, Yi Chern Tan, Xi Victoria Lin, Suyi Li, Heyang Er, Irene Li, Bo Pang, Tao Chen, Emily Ji, Shreya Dixit, David Proctor, Sungrok Shim, Jonathan Kraft, Vincent Zhang, Caiming Xiong, Richard Socher, and

Dragomir Radev. 2019b. SParC: Cross-domain semantic parsing in context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4511–4523, Florence, Italy. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

A Strategy search

Below are the details regarding the process of searching for the best strategy:

- Dealing with long texts, how to trim sequences longer than 512 tokens: first tokens, last tokens, first 128 tokens with the last 382 tokens, first 512 tokens aggregated with last 512 tokens, or first 512 tokens aggregated with middle 512 tokens and last 512 tokens. Best strategy for this step was: first 512 tokens concatenated with the last 512 tokens. This leads to a final representation (after BERT layer) of size 1,536 for *base* model and 2,048 for *large* model. Aggregation methods include concatenation, mean and max pooling.

- Pooling type: <CLS> token, mean or max pooling. Best strategy for this step: <CLS> token.

- BERT-layer unfreezing: training the full model from the first step, training only the classification layers for a number of epochs followed by training the whole model for another number of epochs, gradually unfreezing a number of layers per epochs.

- Learning rate: constant learning rate of 1e-5, 2e-5 or 5e-5, discriminative learning rate with decay factor of 0.95 or 0.90, slanted triangular learning rate with maximum learning rate of 1e-4, 2.5e-5 or 5e-5, cutout fraction of 0.1 and ratio of 32. The best strategy for this step: slanted triangular learning rate with maximum learning rate of 2e-5, cutout 0.1 and ratio of 32.

Parameter	Extra	Mean AUC	Std AUC
Trimming			
first (f)	*	81.01	0.19
last (l)	*	80.31	0.22
middle (m)	*	80.29	0.05
first128 & last382	concat	80.79	0.23
first & last	mean	81.25	0.12
first & last	max	81.05	0.25
first & last	**concat**	**81.36**	**0.13**
first & middle & last	mean	81.18	0.22
first & middle & last	max	81.21	0.17
first & middle & last	concat	81.23	0.13
Pooling type			
<CLS>	*	**81.36**	**0.13**
Mean	*	80.80	0.31
Max	*	80.90	0.28
Layer unfreezing			
full model	*	**81.36**	**0.13**
classification + full	5,5	77.14	0.28
classification + full	3,7	79.98	0.55
gradually unfreezing	2	80.26	0.28
gradually unfreezing	3	80.73	0.33
Learning Rate			
constant	1e-5	81.36	0.13
constant	2e-5	81.31	0.16
constant	5e-5	79.89	1.54
discriminative	1e-5, 0.95	80.70	0.09
discriminative	1e-5, 0.90	79.66	0.17
discriminative	2e-5, 0.95	81.28	0.20
discriminative	2e-5, 0.90	81.23	0.11
discriminative	5e-5, 0.95	80.34	1.68
discriminative	5e-5, 0.90	81.36	0.20
slanted triangular	0.1,1e-4,32	72.91	6.42
slanted triangular	0.1,5e-5,32	79.75	3.31
slanted triangular	**0.1,2.5e-5,32**	**81.45**	**0.21**
Dropout value			
0.1	*	**81.45**	**0.21**
0.25	*	81.33	0.22
0.5	*	81.19	0.25
Fully connected layers			
128	*	81.45	0.21
256,128	*	81.39	0.13
128,64	*	**81.47**	**0.18**
256,128,64	*	81.32	0.14
128,64,32	*	81.43	0.13

Table 6: Detailed results on RoBanking using only the plea of the plaintiff.

- Dropout value applied after BERT-layer: dropout values of 0.1, 0.25 or 0.5. The best value was obtained using a dropout value of 0.1.

- Configuration of fully connected layers after BERT-layer: (256,128) or (128,64) or (256,128,64) or (128,64,32). Best configuration is (128,64).

For more details and results for each individual component, refer to Table 6.

B Baselines Hyperparameters

- SVM with string kernels uses the combination of intersection, presence and spectrum string kernels on 5-7 character n-grams
- CNN with 300 feature maps of length 6, sequence lengths of 800 words, Adam Optimizer, learning rate = 0.001, dropout = 0.3
- BI-LSTM model comprises a BI-LSTM encoder with a global attention mechanism and a fully connected layer with 64 neurons. For the BI-LSTM encoder we used a dropout layer with 0.2 probability, and for the fully connected layer 0.1 dropout probability. The maximum sequence length is set to 800 and the input consists of Word2Vec embeddings of size 100, pretrained on the data reserved for training. We used Adam optimizer with default parameters (0.01 learning rate).